



SKRIPSI

OPTIMASI REST API LARAVEL DENGAN METODE *IN-MEMORY CACHE* DAN *QUEUE* (STUDI KASUS: APLIKASI *JADI JUARA*)

MUHAMMAD RAFLY AMARTHARIZQI

NPM 21081010311

DOSEN PEMBIMBING

Fawwaz Ali Akbar, S.Kom., M.Kom.

Firza Prima Aditiawan, S.Kom., MTI

**KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN JAWA TIMUR
FAKULTAS ILMU KOMPUTER
PROGRAM STUDI INFORMATIKA
SURABAYA
2025**



SKRIPSI

OPTIMASI REST API LARAVEL DENGAN METODE *IN-MEMORY CACHE* DAN *QUEUE* (STUDI KASUS: APLIKASI *JADI JUARA*)

MUHAMMAD RAFLY AMARTHARIZQI
NPM 21081010311

DOSEN PEMBIMBING

Fawwaz Ali Akbar, S.Kom., M.Kom.
Firza Prima Aditiawan, S.Kom., MTI

**KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN JAWA TIMUR
FAKULTAS ILMU KOMPUTER
PROGRAM STUDI INFORMATIKA
SURABAYA
2025**

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

OPTIMASI REST API LARAVEL DENGAN METODE IN- MEMORY CACHE DAN QUEUE (STUDI KASUS: APLIKASI JADI JUARA)

Oleh:

MUHAMMAD RAFLY AMARTHARIZQI

NPM. 21081010311

Telah dipertahankan dihadapan dan diterima oleh Tim Penguji Skripsi Prodi
Informatika Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran
Jawa Timur Pada Tanggal 05 Juni 2025

Fawwaz Ali Akbar, S.Kom, M.Kom

NIP. 19920317 201803 1 002

(Pembimbing I)

Firza Prima Aditiawan, S.Kom., MTI

NIP. 19860523 202121 1 003

(Pembimbing II)

Dr. Ir. Kartini, S.Kom. MT.

NIP. 19611110 199103 2 001

(Penguji I)

Eka Prakarsa Mandyartha, S.T., M.Kom

NIP. 19880525 201803 1 001

(Penguji II)

Mengetahui,

Dekan Fakultas Ilmu Komputer

Prof. Dr. Ir. Novirina Hendrasarie, M.T.

NIP. 1968112 6199403 2 001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

**OPTIMASI REST API LARAVEL DENGAN METODE *IN-MEMORY*
CACHE DAN *QUEUE*
(STUDI KASUS: APLIKASI JADI JUARA)**

Oleh:
MUHAMMAD RAFLY AMARTHARIZQI
NPM. 21081010311



Menyetujui,
Koordinator Program Studi Informatika

Fetty Tri Anggraeny, S.Kom., M.Kom.

NIP. 1982021 12021212 005

Halaman ini sengaja dikosongkan

SURAT PERNYATAAN BEBAS PLAGIASI

Yang bertandatangan dibawah ini:

Nama Mahasiswa : Muhammad Rafly Amartharizqi
NPM : 21081010311
Program Studi : Informatika
Fakultas : Ilmu Komputer

Menyatakan bahwa dalam dokumen ilmiah Skripsi ini tidak terdapat bagian dari karya ilmiah lain yang telah diajukan untuk memperoleh gelar akademik di suatu lembaga Pendidikan Tinggi, dan juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang/lembaga lain, kecuali yang secara tertulis disitasi dalam dokumen ini dan disebutkan secara lengkap dalam daftar pustaka.

Dan saya menyatakan bahwa dokumen ilmiah ini bebas dari unsur-unsur plagiasi. Apabila adikemudian hari ditemukan indikasi plagiat pada Skripsi ini, saya bersedia menerima sanksi sesuai dengan peraturan perundang-undangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya tanpa ada paksaan dari siapapun juga dan untuk dipergunakan sebagaimana mestinya.

Surabaya, 18 Juni 2025

Yang Membuat Pernyataan,



MUHAMMAD RAFLY AMARTHARIZQI

NPM. 21081010311

Halaman ini sengaja dikosongkan

ABSTRAK

Nama Mahasiswa/NPM : Muhammad Rafly Amartharizqi / 21081010311
Judul : Optimasi REST API Laravel Dengan Metode *In-Memory Cache dan Queue* (Studi Kasus: Aplikasi Jadi Juara)
Dosen Pembimbing : 1. Fawwaz Ali Akbar, S.Kom., M.Kom.
2. Firza Prima Aditiawan, S.Kom., MTI.

Optimasi performa REST API aplikasi Jadi Juara dengan menggunakan metode *queue*, *in-memory cache*, dan penggantian web server ke FrankenPHP serta kombinasi FrankenPHP dengan kedua metode tersebut. *Tools* yang digunakan untuk mengimplementasikan *queue* dan *in-memory cache* yaitu Redis. Metode *queue* diimplementasikan pada endpoint API yang dapat dijalankan secara asinkron dan metode *in-memory cache* diimplementasikan pada endpoint API yang sering diakses. Hasil pengujian pada endpoint yang diimplementasikan *queue request per second* meningkat sebesar 905%–1039% dan menurunkan *time per request* sebesar 85%–86,6%. Pada kombinasi metode *queue* dan penggunaan FrankenPHP *request per second* meningkat sebesar 1407%–1778% dan *time per request* turun hingga 89%–89,37%. Penggunaan CPU pada kombinasi *queue* dan FrankenPHP turun antara 33,03% hingga 47,17%. Pada endpoint optimasi *in-memory cache request per second* meningkat hingga 116% dan *time per request* menurun hingga 48%. Sedangkan kombinasi antar metode *in-memory cache* dan FrankenPHP meningkatkan *request per second* hingga 116%, *time per request* hingga 48% dan menurunkan penggunaan CPU hingga 36%. Penggunaan metode *queue* dan *in-memory cache* mengakibatkan penggunaan memory meningkat 3%-7% dan kombinasi metode *queue* dan *in-memory cache* dengan web server FrankenPHP meningkat hingga 8%. Dengan hasil ini kombinasi FrankenPHP dengan metode *in-memory cache* serta *queue* menjadi solusi terbaik untuk meningkatkan performa REST API pada aplikasi Jadi Juara.

Kata Kunci: Optimasi REST API, In-Memory Cache, Queue, FrankenPHP

Halaman ini sengaja dikosongkan

ABSTRACT

Student Name / NPM : Muhammad Rafly Amartharizqi / 21081010311
Thesis Title : *Optimization REST API Laravel With In-Memory Cache and Queue Method (Study Case: Jadi Juara Application)*
Advisor : 1. Fawwaz Ali Akbar, S.Kom., M.Kom.
2. Firza Prima Aditiawan, S.Kom., MTI

Optimization of the REST API performance in the Jadi Juara application by using queueing methods, in-memory caching, and switching the web server to FrankenPHP, as well as combining FrankenPHP with both methods. Redis was used as the tool to implement the queue and in-memory cache. The queue method was applied to API endpoints that can be executed asynchronously, while the in-memory cache method was implemented on frequently accessed API endpoints. Testing results showed that on endpoints using the queue method, the requests per second increased by 905%–1039% and the time per request decreased by 85%–86.6%. When combining the queue method with FrankenPHP, requests per second increased by 1407%–1778% and the time per request dropped by 89%–89.37%. CPU usage for the combination of queue and FrankenPHP decreased by 33.03% to 47.17%. For endpoints optimized with in-memory caching, the requests per second increased by up to 116% and the time per request decreased by up to 48%. Meanwhile, the combination of in-memory caching and FrankenPHP improved requests per second by up to 116%, reduced the time per request by up to 48%, and decreased CPU usage by up to 36%. However, the use of queue and in-memory cache methods increased memory usage by 3%–7%, and the combination of both methods with the FrankenPHP web server increased memory usage by up to 8%. Based on these results, the combination of FrankenPHP with in-memory caching and queueing methods proves to be the most effective solution for improving REST API performance in the Jadi Juara application.

Keywords: *Optimization REST API, In-Memory Cache, Queue, FrankenPHP*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat, karunia, dan kemudahan-Nya sehingga penulis dapat menyelesaikan skripsi berjudul **“Optimasi REST API Laravel dengan Metode In-Memory Cache dan Queue (Studi Kasus: Aplikasi Jadi Juara)”**. Karya ini disusun sebagai salah satu syarat memperoleh gelar Sarjana Komputer pada Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur.

Perjalanan menyelesaikan skripsi ini tidaklah mudah; keraguan, tekanan, dan kelelahan kerap muncul di tengah jalan. Namun, penulis dikelilingi orang-orang luar biasa yang senantiasa memberikan dukungan, bimbingan, dan semangat. Oleh karena itu, dengan segala kerendahan hati penulis menyampaikan terima kasih sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Akhmad Fauzi, M.MT., selaku Rektor Universitas Pembangunan Nasional “Veteran” Jawa Timur.
2. Ibu Prof. Dr. Ir. Novirina Hendrasarie, M.T. selaku Dekan Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.
3. Ibu Fetty Tri Anggraeny, S.Kom., M.Kom. selaku Koordinator Program Studi Informatika Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.
4. Bapak Fawwaz Ali Akbar, S.Kom., M.Kom selaku Pembimbing I yang telah dengan penuh dedikasi memberikan waktu, perhatian, dan pemikiran dalam membimbing serta mengarahkan penulis selama proses penyusunan skripsi ini
5. Bapak Firza Prima Aditiawan, S.Kom., MTI selaku Pembimbing II yang telah menyediakan waktu, energi, dan ide dalam membina serta memberikan arahan kepada penulis selama pelaksanaan tugas akhir ini.
6. Ibu Dr. Ir. Kartini, S.Kom. MT., selaku Penguji I pada skripsi penulis.
7. Bapak Eka Prakarsa Mandyaharta, ST., M.Kom., selaku Penguji II pada skripsi penulis.

8. Seluruh Dosen Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional 'Veteran' Jawa Timur yang telah berbagi ilmu, wawasan, dan pengalaman berharga kepada penulis selama masa studi.
9. Bapak Muhammad Ibrahim Hamdani selaku Direktur PT. Bimbingan Belajar Himalaya yang telah memberikan kesempatan kepada penulis sehingga dapat melaksanakan dan menyelesaikan skripsi ini dengan baik.
10. Kedua orang tua penulis dan keluarga penulis yang selalu memberikan semangat selalu doa, dan dukungan tanpa henti selama proses penyusunan skripsi ini
11. Terima kasih kepada teman-teman penulis yaitu kukur, bagjo, agil, obek, diki, naps, ahil, dido, iqbal, bowo, firhan, mbing, bagus serta seluruh teman-teman informatika angkatan 21 yang telah memberikan dukungan semangat kepada penulis sehingga dapat menyelesaikan skripsi ini dengan baik.

Penulis menyadari bahwa di dalam penyusunan skripsi ini banyak terdapat kekurangan. Untuk itu kritik dan saran yang membangun dari semua pihak sangat diharapkan demi kesempurnaan penulisan skripsi ini. Akhirnya, dengan segala keterbatasan yang penulis miliki semoga laporan ini dapat bermanfaat bagi semua pihak umumnya dan penulis pada khususnya.

Surabaya, 18 Juni 2025

Penulis,

Muhammad Rafly Amartharizqi

DAFTAR ISI

LEMBAR PENGESAHAN	iii
LEMBAR PERSETUJUAN	v
SURAT PERNYATAAN BEBAS PLAGIASI.....	vii
ABSTRAK	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR.....	xix
DAFTAR TABEL.....	xxiii
LAMPIRAN.....	xxix
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Tujuan Penelitian	4
1.4. Manfaat Penelitian	4
1.5. Batasan Masalah.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1. Penelitian Terdahulu	7
2.2. REST API.....	8
2.3. Laravel.....	9
2.4. Load Testing.....	10
2.5. K6.....	12
2.6. In-Memory Caching.....	12
2.7. Queue	13
2.8. Redis.....	14
2.9. FrankenPHP	15
2.10. Docker.....	16
2.11. Prometheus.....	16
2.12. Grafana.....	17
2.13. Aplikasi Jadi Juara	17

BAB III HASIL DAN PEMBAHASAN.....	19
3.1. Metodologi Penelitian.....	19
3.1.1. Identifikasi Masalah	19
3.1.2. Studi Literatur.....	20
3.1.3. Analisis Kebutuhan.....	20
3.1.4. Perencanaan Pengujian	20
3.1.5. Pengujian Performa Sebelum Optimasi	21
3.1.6. Persiapan Lingkungan	21
3.1.7. Implementasi	21
3.1.8. Pengujian Performa Sesudah Optimasi	22
3.1.9. Analisis Perbandingan Performa	22
3.1.10. Pengambilan Kesimpulan	22
3.2. Studi Kasus Aplikasi Jadi Juara.....	23
3.2.1 Arsitektur Back-end Aplikasi Jadi Juara	25
3.2.2 Endpoint REST API	26
3.3. Spesifikasi Perangkat.....	27
3.4. Arsitektur pada Redis	28
3.5. Arsitektur FrankenPHP.....	29
3.6. Parameter Optimasi	30
3.7. Rencana Pengujian	31
3.7.1 Rencana Pengujian <i>Request per Second</i>	34
3.7.2. Rencana Pengujian <i>Time per Request</i>	40
3.7.3 Rencana Pengujian Penggunaan CPU	45
3.7.4. Rencana Pengujian Penggunaan Memori	51
BAB IV HASIL DAN PEMBAHASAN	59
4.1. Lingkungan Pengujian.....	59
4.1.1 Perangkat Keras (Hardware)	59
4.1.2. Perangkat Lunak (Software).....	60
4.1.3. Jaringan Internet	61
4.1.4. Konfigurasi Docker	61
4.1.5. K6	75

4.2.	Implementasi Kode Program.....	77
4.2.1	Implementasi <i>Queue</i>	78
4.2.2	Implementasi <i>In-Memory Caching</i>	84
4.3.	Hasil Pengujian	92
4.3.1	Pengujian Sebelum Dioptimasi	93
4.3.2	Pengujian Setelah Optimasi Dengan Redis	115
4.3.3	Pengujian Setelah Optimasi Dengan FrankenPHP	138
4.3.4	Pengujian Setelah Optimasi Dengan FrankenPHP dan Redis.....	158
4.4.	Analisis Perbandingan Setiap Endpoint	180
4.4.1.	Perbandingan Hasil Pengujian Request per Second.....	180
4.4.2.	Perbandingan Hasil Pengujian Time per Request.....	197
4.4.3.	Perbandingan Hasil Pengujian Penggunaan CPU	215
4.4.4.	Perbandingan Hasil Pengujian Penggunaan Memory	235
4.5.	Analisis Perbandingan Secara Rata-Rata	254
4.5.1.	Rata-rata Endpoint Optimasi Queue	254
4.5.2.	Rata-rata Endpoint Optimasi In-Memory Cache	258
4.6.	Kesimpulan Uji Statistik Untuk Pengembang.....	262
BAB V	PENUTUP	265
5.1.	Kesimpulan	265
5.2.	Saran.....	267
DAFTAR PUSTAKA		269
LAMPIRAN.....		273

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur REST API	9
Gambar 2. 2 Arsitektur MVC.....	9
Gambar 2. 3 Arsitektur MVC dengan Routing	10
Gambar 2. 4 Algoritma <i>In-Memory Caching</i>	12
Gambar 2. 5 Interaksi Asinkron <i>Client</i> dan <i>Worker</i>	14
Gambar 3. 1 Alur Metodologi Penelitian.....	19
Gambar 3. 2 Tampilan Aplikasi Jadi Juara (1)	23
Gambar 3. 3 Tampilan Aplikasi Jadi Juara (2)	24
Gambar 3. 4 Tampilan Aplikasi Jadi Juara (3)	24
Gambar 3. 5 Alur <i>Back-end</i> Aplikasi Jadi Juara	25
Gambar 3. 6 Arsitektur Redis untuk Implementasi Queue	28
Gambar 3. 7 Arsitektur Redis untuk Implementasi <i>In-memory Caching</i>	28
Gambar 3. 8 Arsitektur FrankenPHP	29
Gambar 3. 9. Alur Pengujian Setelah Optimasi	31
Gambar 4. 1. Hasil Pengujian <i>Request per Second</i> dan <i>Time per Request</i> Sebelum Dioptimasi	114
Gambar 4. 2. Hasil Pengujian Penggunaan CPU dan Memory Sebelum Dioptimasi	114
Gambar 4. 3. Hasil Pengujian <i>Request per Second</i> dan <i>Time per Request</i> Setelah Diimplementasikan Redis	137
Gambar 4. 4. Hasil Pengujian Penggunaan CPU dan Memory Setelah Diimplementasikan Redis	137
Gambar 4. 5. Hasil Pengujian <i>Request per Second</i> dan <i>Time per Request</i> Setelah Menggunakan FrankenPHP	158
Gambar 4. 6. Hasil Pengujian Penggunaan CPU dan Memory Setelah Menggunakan FrankenPHP	158
Gambar 4. 7. Hasil Pengujian <i>Request per Second</i> dan <i>Time per Request</i> Setelah Diimplementasikan Redis dan Menggunakan FrankenPHP	179

Gambar 4. 8, Hasil Pengujian Penggunaan CPU dan Memory Setelah Diimplementasikan Redis dan FrankenPHP.....	179
Gambar 4. 9. Grafik Request per Second Pada Endpoint /register-student.....	181
Gambar 4. 10. Grafik Request per Second Pada Endpoint /forgot-password	182
Gambar 4. 11. Grafik Request per Second Pada Endpoint /resend-otp.....	183
Gambar 4. 12. Grafik Request per Second Pada Endpoint /finish-exam	184
Gambar 4. 13. Grafik Request per Second Endpoint /top-3-banner.....	187
Gambar 4. 14. Grafik Request per Second Endpoint /subject	188
Gambar 4. 15. Grafik Request per Second Endpoint /top-5-news	189
Gambar 4. 16. Grafik Request per Second Pada Endpoint /top-5-events.....	190
Gambar 4. 17. Grafik Request per Second Pada Endpoint /event-list.....	191
Gambar 4. 18. Grafik Request per Second Pada Endpoint /events-detail	192
Gambar 4. 19. Grafik Request per Second Pada Endpoint /news-list	193
Gambar 4. 20. Grafik Request per Second Endpoint /news-detail	194
Gambar 4. 21. Grafik Request per Second Endpoint /grade	195
Gambar 4. 22. Grafik Request per Second Endpoint /exam-assignment	196
Gambar 4. 23. Grafik Time per Request Pada Endpoint /register-student.....	198
Gambar 4. 24. Grafik Time per Request Pada Endpoint /forgot-password.....	199
Gambar 4. 25. Grafik Time per Request Pada Endpoint /resend-otp	200
Gambar 4. 26. Grafik Time per Request Pada Endpoint /finish-exam.....	201
Gambar 4. 27. Grafik Time per Request Endpoint /top-3-banner	205
Gambar 4. 28. Grafik Time per Request Endpoint /subject	206
Gambar 4. 29. Grafik Time per Request Pada Endpoint /top-5-news	207
Gambar 4. 30. Grafik Time per Request Pada Endpoint /top-5-events	208
Gambar 4. 31. Grafik Time per Request Pada Endpoint /events-list	209
Gambar 4. 32. Grafik Time per Request Pada Endpoint /events-detail	210
Gambar 4. 33. Grafik Time per Request Pada Endpoint /news-list	211
Gambar 4. 34. Grafik Time per Request Pada Endpoint /news-detail	212
Gambar 4. 35. Grafik Time per Request Pada Endpoint /grade	213
Gambar 4. 36. Grafik Time per Request Pada Endpoint /exam-assignment.....	214
Gambar 4. 37. Grafik Penggunaan CPU Endpoint /register-student.....	216

Gambar 4. 38. Grafik Penggunaan CPU Pada Endpoint /forgot-password	217
Gambar 4. 39. Grafik Penggunaan CPU Pada Endpoint /resend-otp.....	219
Gambar 4. 40. Grafik Penggunaan CPU Pada Endpoint /finish-exam	220
Gambar 4. 41. Grafik Penggunaan CPU Pada Endpoint /top-3-banner.....	223
Gambar 4. 42. Grafik Penggunaan CPU Pada Endpoint /subject	225
Gambar 4. 43. Grafik Penggunaan CPU Pada Endpoint /top-5-news.....	226
Gambar 4. 44. Grafik Penggunaan CPU Pada Endpoint /top-5-events.....	227
Gambar 4. 45. Grafik Penggunaan CPU Pada Endpoint /events-list	228
Gambar 4. 46. Grafik Penggunaan CPU Pada Endpoint /events-detail	229
Gambar 4. 47. Grafik Penggunaan CPU Pada Endpoint /news-list	230
Gambar 4. 48. Grafik Penggunaan CPU Pada Endpoint /news-detail	231
Gambar 4. 49. Grafik Penggunaan CPU Pada Endpoint /grade.....	232
Gambar 4. 50. Grafik Penggunaan CPU Pada Endpoint /exa-assignment.....	234
Gambar 4. 51. Grafik Penggunaan Memory Pada Endpoint /register-student....	236
Gambar 4. 52. Grafik Penggunaan Memory Pada Endpoint /forgot-password ..	237
Gambar 4. 53. Grafik Penggunaan Memory Pada Endpoint /resend-otp.....	238
Gambar 4. 54. Grafik Penggunaan Memory Pada Endpoint /finish-exam	239
Gambar 4. 55. Grafik Penggunaan memory Pada Endpoint /top-3-banner	243
Gambar 4. 56. Grafik Penggunaan Memory Pada Endpoint /subject	244
Gambar 4. 57. Grafik Penggunaan Memory Pada Endpoint /top-5-news.....	245
Gambar 4. 58. Grafik Penggunaan Memory Pada Endpoint /top-5-events.....	246
Gambar 4. 59. Grafik Penggunaan Memory Pada Endpoint /events-list.....	247
Gambar 4. 60. Grafik Penggunaan Memory Pada Endpoint /events-detail	248
Gambar 4. 61. Grafik Penggunaan Memory Pada Endpoint /news-list	250
Gambar 4. 62. Grafik Penggunaan Memory Pada Endpoint /news-detail	251
Gambar 4. 63. Grafik Penggunaan Memory Pada Endpoint /grade.....	252
Gambar 4. 64. Grafik Penggunaan Memory Pada Endpoint /exam-assignment.	253

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 3. 1 Endpoint API untuk implementasi Queue	26
Tabel 3. 2 Endpoint API untuk Implementasi In-memory Cache.....	26
Tabel 3. 3 Spesifikasi Server.....	27
Tabel 3. 4 Spesifikasi Server Saat Implementasi FrankenPHP.....	27
Tabel 3. 5. Parameter Load Testing	30
Tabel 3. 6. Rencana Pengujian <i>Load Testing Queue</i>	32
Tabel 3. 7. Rencana Pengujian Load Testing <i>In-memory Cache</i>	33
Tabel 3. 8 Test Case Parameter <i>Request per Second</i> Implementasi <i>Queue</i>	34
Tabel 3. 9 Test Case Parameter Request per Second Implementasi In-memory Caching	35
Tabel 3. 10 Test Case Parameter Time per Request Implementasi Queue	40
Tabel 3. 11 Test Case Parameter Time per Request Implementasi In-memory caching	41
Tabel 3. 12 Test Case Penggunaan CPU Pada Implementasi <i>Queue</i>	46
Tabel 3. 13. Test case Penggunaan CPU Pada Implementasi In-memory caching	47
Tabel 3. 14 Test Case Penggunaan Memori Pada Implementasi <i>Queue</i>	51
Tabel 3. 15 Test Case Penggunaan Memori Pada Implementasi <i>In-memory Caching</i>	53
Tabel 4. 1 Spesifikasi Perangkat Keras (Hardware)	59
Tabel 4. 2 Spesifikasi Perangkat Lunak (Software).....	60
Tabel 4. 3. Konfigurasi docker-compose Nginx dan PHP-FPM.....	61
Tabel 4. 4. Konfigurasi Dockerfile Nginx dan PHP-FPM	62
Tabel 4. 5. Konfigurasi Nginx pada Lingkungan Pengujian Nginx dan PHP-FPM	64
Tabel 4. 6. Konfigurasi docker-compose Nginx, PHP-FPM dan Redis.....	65
Tabel 4. 7. Konfigurasi Dockerfile Nginx, PHP-FPM, dan Redis	66
Tabel 4. 8. Konfigurasi Nginx.conf Pada Lingkungan Pengujian Nginx, PHP-FPM dan Redis	67
Tabel 4. 9. Konfigurasi docker-compose FrankenPHP.....	69

Tabel 4. 10. Konfigurasi Dockerfile FrankenPHP	70
Tabel 4. 11. Konfigurasi docker-compose FrankenPHP dan Redis	71
Tabel 4. 12. Konfigurasi Dockerfile FrankenPHP dan Redis.....	72
Tabel 4. 13. Konfigurasi Pemantauan Penggunaan CPU dan Memory.....	73
Tabel 4. 14. Konfigurasi prometheus.yml Untuk Monitoring	75
Tabel 4. 15. Kode Testing K6.....	75
Tabel 4. 16 Konfigurasi .env Laravel	77
Tabel 4. 17. Implementasi Pada Endpoint /register-student.....	78
Tabel 4. 18. Implementasi Pada Endpoint /forgot-password.....	79
Tabel 4. 19. Implementasi Pada Endpoint /resend-otp.....	80
Tabel 4. 20. Implementasi Pada Endpoint /finish-exam.....	80
Tabel 4. 21. Jobs Pada ProcessFinishExam.php.....	82
Tabel 4. 22 Implementasi Pada Endpoint /top-3-banner	84
Tabel 4. 23 Implementasi Pada Endpoint /subject	85
Tabel 4. 24 Implementasi Pada Endpoint /top-5-news.....	86
Tabel 4. 25 Implementasi Pada Endpoint /top-5-events.....	87
Tabel 4. 26. Implementasi Pada Endpoint /exam-assignment.....	88
Tabel 4. 27 Implementasi Pada Endpoint /news.....	89
Tabel 4. 28. Implementasi Pada Endpoint /news-detail	89
Tabel 4. 29 Implementasi Pada Endpoint /events-list	90
Tabel 4. 30. Implementasi Pada Endpoint /events-detail	91
Tabel 4. 31. Implementasi Pada Endpoint /grade.....	92
Tabel 4. 32. Hasil Pengujian Request per Second Sebelum Dioptimasi Dengan Queue.....	93
Tabel 4. 33. Hasil Pengujian Time per Request Sebelum Dioptimasi Dengan Queue	94
Tabel 4. 34. Hasil Pengujian Penggunaan CPU Sebelum Dioptimasi Dengan Queue	96
Tabel 4. 35. Hasil Pengujian Penggunaan Memory Sebelum Dioptimasi Dengan Queue.....	97

Tabel 4. 36. Hasil Pengujian Request per Second Sebelum Dioptimasi In-Memory Cache.....	98
Tabel 4. 37. Hasil Pengujian Time per Request Sebelum Dioptimasi In-memory Cache.....	101
Tabel 4. 38. Hasil Pengujian Penggunaan CPU sebelum Dioptimasi In-Memory Cache.....	104
Tabel 4. 39. Hasil Pengujian Penggunaan Memory Sebelum Dioptimasi In-Memory Cache.....	109
Tabel 4. 40. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan Queue	115
Tabel 4. 41. Hasil Pengujian Time per Request Setelah Dioptimasi Dengan Queue	116
Tabel 4. 42. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan Queue	117
Tabel 4. 43. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan Queue	118
Tabel 4. 44. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan In-Memory Cache.....	120
Tabel 4. 45. Hasil Pengujian Time per Request Setelah Dioptimasi Dengan In-Memory Cache.....	123
Tabel 4. 46. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan In-Memory Cache.....	127
Tabel 4. 47. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan In-Memory Cache.....	132
Tabel 4. 48. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan FrankenPHP Tanpa Queue.....	138
Tabel 4. 49. Hasil Pengujian Time per Request Setelah Dioptimasi Dengan FrankenPHP Tanpa Queue.....	139
Tabel 4. 50. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan FrankenPHP Tanpa Queue.....	140

Tabel 4. 51. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan FrankenPHP Tanpa Queue	141
Tabel 4. 52. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan FrankenPHP Tanpa In-Memory Cache	143
Tabel 4. 53. Hasil Pengujian Time per Request Setelah Dioptimasi Dengan FrankenPHP Tanpa In-Memory Cache	146
Tabel 4. 54. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan FrankenPHP Tanpa In-Memory Cache	149
Tabel 4. 55. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan FrankenPHP Tanpa In-Memory Cache	153
Tabel 4. 56. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan FrankenPHP dan Queue.....	159
Tabel 4. 57. Hasil Pengujian Time per Request Setelah Dioptimasi Dengan FrankenPHP dan Queue.....	160
Tabel 4. 58. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan FrankenPHP dan Queue.....	161
Tabel 4. 59. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan FrankenPHP dan Queue.....	162
Tabel 4. 60. Hasil Pengujian Request per Second Setelah Dioptimasi Dengan FrankenPHP dan In-memory Cache	164
Tabel 4. 61. Hasil Pengujian Time per Request Setelah Dioptimasi FrankenPHP dan In-memory Cache	167
Tabel 4. 62. Hasil Pengujian Penggunaan CPU Setelah Dioptimasi Dengan FrankenPHP dan In-Memory Cache.....	170
Tabel 4. 63. Hasil Pengujian Penggunaan Memory Setelah Dioptimasi Dengan FrankenPHP dan In-Memory Cache.....	174
Tabel 4. 64. Hasil Pengujian Request per Second Pada Optimasi Dengan Queue	180
Tabel 4. 65. Perbandingan Hasil Pengujian Request per Second Pada Optimasi In-Memory Cache	185
Tabel 4. 66. Perbandingan Time per Request Pada Optimasi Queue.....	197

Tabel 4. 67. Perbandingan Hasil Pengujian Time per Request.....	202
Tabel 4. 68. Perbandingan Hasil Pengujian Penggunaan CPU Optimasi Queue	215
Tabel 4. 69. Perbandingan Hasil Pengujian Penggunaan CPU Optimasi In-Memory Cache.....	221
Tabel 4. 70. Perbandingan Hasil Pengujian Penggunaan Memory Optimasi Queue	235
Tabel 4. 71. Perbandingan Hasil Pengujian Penggunaan Memory Optimasi In- Memory Cache.....	240
Tabel 4. 72. Rata-rata Request per Second Optimasi Queue	254
Tabel 4. 73. Rata-rata Time per Request Optimasi Queue	256
Tabel 4. 74. Rata-rata Penggunaan CPU Optimasi Queue.....	256
Tabel 4. 75. Rata-rata Penggunaan Memory Optimasi Queue.....	257
Tabel 4. 76. Rata-rata Request per Second Optimasi In-Memory Cache	258
Tabel 4. 77. Rata-rata Time per Request Optimasi In-Memory Cache	259
Tabel 4. 78. Rata-rata Penggunaan CPU Optimasi In-Memory Cache.....	260
Tabel 4. 79. Rata-rata Penggunaan Memory Optimasi In-Memory Cache.....	261

Halaman ini sengaja dikosongkan

LAMPIRAN

Lampiran 1. Hasil Pengujian.....	273
Lampiran 2. Surat Implementasi Kerja Sama	277
Lampiran 3. Surat Kerja Sama	280

Halaman ini sengaja dikosongkan